

Experiment – 2

Name: Muskan Soni

UID: 22BCS16851

Branch: BE-CSE

Section/Group: DL-902/B

Semester: 6th

Date:

Subject: Project Based Learning in java

Subject Code: 22CSH-359

1. Aim: Design and implement a simple inventory control system for a small video rental store.

2. Implementation:

```
import java.util.HashMap;
import java.util.Scanner;

public class VideoRentalSystem {

    private HashMap<String, Integer> inventory; // Title -> Quantity

    public VideoRentalSystem() {
        inventory = new HashMap<>();
        inventory.put("The Shawshank Redemption", 5);
        inventory.put("The Godfather", 3);
        inventory.put("The Dark Knight", 7);
        inventory.put("Pulp Fiction", 2);
    }

    public void addVideo(String title, int quantity) {
        if (inventory.containsKey(title)) {
            inventory.put(title, inventory.get(title) + quantity);
        } else {
            inventory.put(title, quantity);
        }
        System.out.println(quantity + " copies of " + title + " added to inventory.");
    }
}
```

```
public void rentVideo(String title) {
    if (inventory.containsKey(title)) {
        int quantity = inventory.get(title);
        if (quantity > 0) {
            inventory.put(title, quantity - 1);
            System.out.println("Rented " + title + ".");
        } else {
            System.out.println("Sorry, " + title + " is currently out of stock.");
        }
    } else {
        System.out.println("Sorry, " + title + " is not in our inventory.");
    }
}

public void returnVideo(String title) {
    if (inventory.containsKey(title)) {
        inventory.put(title, inventory.get(title) + 1);
        System.out.println("Returned " + title + ".");
    } else {
        System.out.println("Error: Video not found in the system."); // Should not happen ideally
    }
}

public void displayInventory() {
    System.out.println("\nCurrent Inventory:");
    for (String title : inventory.keySet()) {
        System.out.println("'" + title + "': " + inventory.get(title) + " copies");
    }
}

public static void main(String[] args) {
    VideoRentalSystem system = new VideoRentalSystem();
}
```

```
Scanner scanner = new Scanner(System.in);

while (true) {

    System.out.println("\nVideo Rental System Menu:");

    System.out.println("1. Add Video");
    System.out.println("2. Rent Video");
    System.out.println("3. Return Video");
    System.out.println("4. Display Inventory");
    System.out.println("5. Exit");

    System.out.print("Enter your choice: ");

    int choice = scanner.nextInt();

    scanner.nextLine(); // Consume newline

    switch (choice) {

        case 1:

            System.out.print("Enter video title: ");

            String addTitle = scanner.nextLine();

            System.out.print("Enter quantity: ");

            int addQuantity = scanner.nextInt();

            scanner.nextLine(); // Consume newline

            system.addVideo(addTitle, addQuantity);

            break;

        case 2:

            System.out.print("Enter video title to rent: ");

            String rentTitle = scanner.nextLine();

            system.rentVideo(rentTitle);

            break;

        case 3:

            System.out.print("Enter video title to return: ");

            String returnTitle = scanner.nextLine();

            system.returnVideo(returnTitle);

            break;
```

case 4:

```
system.displayInventory();  
  
break;
```

case 5:

```
System.out.println("Exiting...");  
  
scanner.close();  
  
return;
```

default:

```
System.out.println("Invalid choice. Please try again.");
```

```
}
```

```
}
```

```
}
```

```
}
```

3. Output:

| PROBLEMS | 18 | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS | SPELL CHECKER | 1 | COMMENTS |
|----------|----|--------|---------------|----------|-------|---------------|---|----------|
|----------|----|--------|---------------|----------|-------|---------------|---|----------|

```
Video Rental System Menu:  
1. Add Video  
2. Rent Video  
3. Return Video  
4. Display Inventory  
5. Exit  
Enter your choice: 1  
Enter video title: Avengers  
Enter quantity: 2  
2 copies of 'Avengers' added to inventory.  
  
Video Rental System Menu:  
1. Add Video  
2. Rent Video  
3. Return Video  
4. Display Inventory  
5. Exit  
Enter your choice: 2  
Enter video title to rent: Avengers  
Rented 'Avengers'.  
  
Video Rental System Menu:  
1. Add Video  
2. Rent Video  
3. Return Video  
4. Display Inventory  
5. Exit  
Enter your choice: 3  
Enter video title to return: Avengers  
Returned 'Avengers'.  
  
Video Rental System Menu:  
1. Add Video  
2. Rent Video  
3. Return Video  
4. Display Inventory  
5. Exit  
Enter your choice: 5  
Exiting...
```

4. Learning outcomes:

- **Data Storage:** Using HashMaps to manage inventory data efficiently.
- **OOP Concepts:** Designing a class to represent the video rental system.
- **Menu-Driven Programs:** Creating interactive programs with user menus.
- **Basic Input/Output:** Handling user input and displaying output.
- **Conditional Logic:** Implementing logic for renting, returning, and managing inventory.